AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method for verifying ~~a generated~~ computer code having a ~~first~~ plurality of lines generated by a code generating module from a model file of a system including a plurality of functions generated by a model module, the method comprising:

processing the model file, by a verification module, to determine ~~generate an expected computer code having a second plurality of lines from the model file, each line in the second plurality of lines corresponding to a line in the first plurality of lines~~ values, inputs, outputs, function type, and syntax for each of the plurality of functions;

determining expected code for the generated computer code based on the determined values, inputs, outputs, function type, and syntax for the generated computer code;

comparing each line ~~in the first plurality of lines~~ of the generated computer code and ~~each corresponding line in the second plurality of lines~~ the expected code to determine if the generated computer code ~~and the expected computer code match~~ includes correct values, correct inputs, correct outputs, correct functions, and correct syntax; and

transmitting an error message if one or more lines of the generated computer code ~~and the expected computer code do not match~~ does not include a correct value, a correct input, a correct output, a correct function, or a correct syntax based on the comparison.

2. (Currently Amended) The method of claim 1 further, comprising the steps of:

comparing each line ~~in the first plurality of lines and each corresponding line in the second plurality of lines~~ of the generated computer code and the expected code to determine if the ~~first plurality of lines~~ generated computer code is missing a line of code; and ~~does not include a line of code included in the second plurality of lines; and~~

transmitting the error message if the ~~first plurality of lines does not include the line of code included in the second plurality of lines~~ if the generated computer code is missing the line of code.

3. (Currently Amended) The method of claim 1, further comprising the steps of:

comparing each line ~~in the first plurality of lines and each corresponding line in the second plurality of lines~~ of the generated computer code and the expected code to determine if the ~~first plurality of lines~~ generated computer code includes ~~any line of code not in the second plurality of lines~~ an extraneous line of code; and

transmitting the error message if the ~~first plurality of lines includes any line of code not in the second plurality of lines~~ the generated computer code includes the extraneous line of code.

4. (Currently Amended) The method of claim 1, further comprising the steps of:

comparing each line ~~in the first plurality of lines and each corresponding line in the second plurality of lines~~ of the generated computer code and the expected code to determine if the ~~first plurality of lines are~~ the generated computer code is in a ~~similar~~ logical order ~~as the second plurality of lines~~; and

transmitting the error message if the ~~first plurality of lines are~~ generated computer code is not in the ~~similar~~ logical order.

5. (Currently Amended) The method of claim 1, further comprising the steps of:

comparing a first header information section in the generated computer code ~~first plurality of lines and a second header information section in the second plurality of lines~~ and second header information section in the expected code to determine if the first header information section matches the second header information section; and

transmitting the error message if the first header information section does not match the second header information section.

6. (Currently Amended) The method of claim 1, further comprising the steps of:

comparing a first declared variable section in the ~~first plurality of lines to~~ generated computer code and a second declared variable section in the ~~second plurality of lines~~ expected code to determine if the first declared variable section matches the second declared variable section; and

transmitting the error message if the first declared variable section does not match the second declared variable section.

7. (Currently Amended) A computer-readable storage medium containing a set of instructions for verifying a generated computer code having a ~~first~~ plurality of lines from a code generating module, the generated computer code automatically generated from a model file of a system having a plurality of functions and created by a model module, the set of instructions comprising:

code that reads in the model file;

code that determines values, inputs, outputs, function type, and syntax for each of the plurality of functions in the generated computer code;

code that generates an expected computer code ~~having a second plurality of lines from~~ based on ~~the model file, each line in the second plurality of lines corresponding with a line in the first plurality of lines~~ the determined values, inputs, outputs, function type, and syntax;

code that reads in the generated computer code;

code that compares each line in the ~~first plurality of lines and each corresponding line in the second plurality of lines~~ generated computer code and the expected code to determine if the generated computer code ~~and the~~ includes the determined values, inputs, outputs, function type, and syntax in the expected computer code ~~match~~; and

code that transmits an error message if one or more lines in the generated computer code ~~and the expected computer code do not match~~ does not include a determined value, a determined input, a determined output, a determined function, or a determined syntax based on the comparison.

8. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the ~~first plurality of lines~~ generated computer code and ~~each corresponding line in the second plurality of lines~~ the expected code; and

code that transmits the error message if one of the ~~first~~ plurality of lines does not include ~~all of the lines of the second plurality of lines~~ the determined value, the determined input, the determined output, the determined function, the determined syntax, or combinations thereof.

9. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the ~~first~~ plurality of lines and ~~each corresponding line in the second plurality of lines~~ the expected code to determine if the ~~first~~ plurality of lines includes ~~any~~ a line of code ~~not in the second plurality of lines~~ that is not determined in the expected code; and

code that transmits the error message if the generated computer code includes any line of code not determined in the expected computer code.

10. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the ~~first~~ plurality of lines and ~~each corresponding line in the second plurality of lines~~ the expected code to determine if the ~~first~~ plurality of lines are in a correct ~~similar~~ logical order ~~as the second plurality of lines~~; and

code that transmits the error message if the ~~first~~ plurality of lines are not in the correct ~~similar~~ logical order.

11. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

code that compares a first header information section in the ~~first plurality of lines~~ generated computer code and a second header information section in the ~~second plurality of lines~~ expected code to determine if the first header information section matches the second header information section; and

code that transmits the error message if the first header information section does not match the second header information section.

12. (Currently Amended) A system for verifying the contents of a generated computer code having a plurality of lines generated by a code generating module from a model file including a plurality of functions generated by a model module, comprising:

a processor operable to:

process the model file to determine values, inputs, outputs, function type, and syntax for each of the plurality of functions,

~~generate an expected computer code from the model file, each line of code in the expected computer code corresponding to a line of code in the generated computer code,~~

determine expected computer code for the generated computer code based on the determined values, inputs, outputs, functions type, and syntax for the generated computer code,

compare each line in the generated computer code with ~~each corresponding line in~~ the expected computer code to determine if the generated computer code includes correct values, correct inputs, correct outputs, correct functions, and correct syntax, and

transmit an error message if ~~each line~~ one or more lines in the generated computer ~~code and each corresponding line in the expected computer code do not match~~ does not include a correct value, a correct input, a correct output, a correct function, or a correct syntax based on the comparison; and

a display configured to display the error message, the display coupled to the processor.

13. (Currently Amended) The system of claim 12, wherein the error message indicates if ~~each~~ a line is missing in the generated computer code. ~~has all of the content of each corresponding line in the expected computer code.~~

14. (Currently Amended) The system of claim 12, wherein the error message indicates if ~~each~~ a line of the generated computer code has any additional content. ~~not found in each corresponding line of the expected computer code.~~

15. (Currently Amended) The system of claim 12, wherein the processor is operable to compare each line in the ~~first plurality of lines and each corresponding line in the second plurality of lines~~ expected code to determine if the ~~first~~ plurality of lines are in an expected form, and transmit the error message if ~~each~~ one or more of the lines of code in the ~~first~~ plurality of lines do not match the expected form.

16. (Currently Amended) The system of claim 12, wherein the processor is operable to compare each line in the ~~first plurality of lines and each corresponding line in the second plurality of lines~~ expected code to determine if the ~~first~~ plurality of lines includes ~~any line~~ one or more lines of extraneous code. ~~not in the second plurality of lines, and transmit the error message if any lines of code in the first plurality of lines is not in the second plurality of lines.~~

17. (Currently Amended) The system of claim 12, wherein the processor is operable to compare each line in the ~~first plurality of lines and each corresponding line in the second plurality of lines~~ expected code to determine if each line ~~in the first plurality of lines are~~ is in a logical order, and transmit the error message if any line in the ~~first~~ plurality of lines is not in logical order.

18. (Currently Amended) The system of claim 12, wherein the processor is operable to compare a first header information section in the ~~first plurality of lines~~ generated computer code to a second header information section in the ~~second plurality of lines~~ expected computer code to determine if the first header information section matches the second header information section, and transmit the error message if the first header information section does not match the second header information section.

19-20. (Canceled)

21. (Currently Amended) The method of claim 1, further comprising the steps of:

comparing each line in the ~~first plurality of lines and each corresponding line in the second plurality of lines~~ generated computer code and the expected computer code to determine if the ~~first~~ plurality of lines ~~includes all of the lines of the second plurality of lines~~ are complete; and

transmitting an error message if the ~~first~~ plurality of lines ~~does not include all of the lines of the second plurality of lines~~ are incomplete.

22. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the ~~first plurality of lines and each corresponding line in the second plurality of lines~~ generated computer code to the expected computer code to determine if the generated computer code ~~includes all the lines of the expected computer code~~ is complete; and

code that transmits the error message if the ~~first plurality of lines does not include all of the lines of the second plurality of lines~~ generated computer code is incomplete.